

1. A method for describing a service of a device or object in a computing system, comprising:

5 Description Language (IDL) that one to one maps a type of a particular type-based system to an XML schema and vice versa.

2. A method according to claim 1, wherein the XML-based IDL is Type Description Language (TDL).

3. A method according to claim 2, wherein TDL enables a one to one mapping from a programming construct to an XML schema for describing the programming construct and vice versa.

4. A method according to claim 3, wherein the programming construct is one of a pointer programming construct, primitive type programming construct, struct programming construct, class programming construct, array programming construct, subtype programming construct, enumeration type programming construct, service reference construct and bit field programming construct.

5. A method according to claim 2, wherein TDL enables a one to one mapping from a constant value of complex type to an XML schema for describing the constant value of complex type and vice versa.

6. A method according to claim 2, wherein TDL enables a one to one mapping from at least one of properties, methods and events of the type system to an XML schema for describing the at least one of properties, methods and events and vice versa.

7. A method according to claim 3, wherein TDL supports inheritance of programming constructs.
8. A method according to claim 1, wherein the XML-based IDL is the wire format for message communications relating to the service between devices of the computing system.
9. A method according to claim 8, wherein the XML-based IDL enables a one to one mapping from the wire format to the message communications and vice versa.
10. A method according to claim 2, wherein TDL enables the transfer of a service reference across an application boundary.
11. A method according to claim 1, wherein the computing system is a peer to peer distributed computing environment.
12. A method according to claim 1, wherein the XML-based IDL is extendable to map additional constructs of a richer type system to an XML schema and vice versa.
13. A computer readable medium having stored thereon a plurality of computer-executable instructions for performing the method of claim 1.
14. A modulated data signal carrying computer executable instructions for performing the method of claim 1.
15. A computing device comprising means for performing the method of claim 1.
16. A computer readable medium having stored thereon a plurality of computer-executable modules, the computer executable modules, comprising:

a mapping mechanism for describing a service of one of a device and object in a computing system with an extensible markup language (XML)-based Interface Description Language (IDL) that one to one maps a type of a particular type-based system to an XML schema and vice versa.

5

17. A computer readable medium according to claim 16, wherein the XML-based IDL is Type Description Language (TDL).

18. A computer readable medium according to claim 17, wherein TDL enables a one to one mapping from a programming construct to an XML schema for describing the programming construct and vice versa.

19. A computer readable medium according to claim 18, wherein the programming construct is one of a pointer programming construct, primitive type programming construct, struct programming construct, class programming construct, array programming construct, subtype programming construct, enumeration type programming construct, service reference construct and bit field programming construct.

20. A computer readable medium according to claim 17, wherein TDL enables a one to one mapping from a constant value of complex type to an XML schema for describing the constant value of complex type and vice versa.

21. A computer readable medium according to claim 17, wherein TDL enables a one to one mapping from at least one of properties, methods and events of the type system to an XML schema for describing the at least one of properties, methods and events and vice versa.

22. A computer readable medium according to claim 18, wherein TDL supports inheritance of programming constructs.

23. A computer readable medium according to claim 16, wherein the XML-based IDL is the wire format of message communications relating to the service between devices of the computing system.

5

24. A computer readable medium according to claim 23, wherein the XML-based IDL enables a one to one mapping from the wire format to the message communications and vice versa.

25. A computer readable medium according to claim 17, wherein TDL enables the transfer of a service reference across an application boundary.

26. A computer readable medium according to claim 16, wherein the computing system is a peer to peer distributed computing environment.

27. A computer readable medium according to claim 16, wherein the mapping mechanism for the XML-based IDL is extendable to map additional constructs of a richer type system to an XML schema and vice versa.

28. A modulated data signal carrying computer executable instructions output as a result of the execution of the plurality of computer-executable instructions of the computer readable medium of claim 16.

29. A computing device comprising means for carrying out the plurality of computer-executable instructions of the computer readable medium of claim 16.

30. A computing device, comprising:
an operating system, stored as computer-executable instructions on a computer readable

medium, the computer-executable instructions of the operating system comprising:

a mapping mechanism for describing a service of one of a device and object in a computing system with an extensible markup language (XML)-based Interface Description Language (IDL) that one to one maps a type of a particular type-based system to an XML schema and vice versa.

31. A computing device according to claim 30, wherein the XML-based IDL is Type Description Language (TDL).

32. A computing device according to claim 31, wherein TDL enables a one to one mapping from a programming construct to an XML schema for describing the programming construct and vice versa.

33. A computing device according to claim 32, wherein the programming construct is one of a pointer programming construct, primitive type programming construct, struct programming construct, class programming construct, array programming construct, subtype programming construct, enumeration type programming construct, service reference construct and bit field programming construct.

34. A computing device according to claim 31, wherein TDL enables a one to one mapping from a constant value of complex type to an XML schema for describing the constant value of complex type and vice versa.

35. A computing device according to claim 31, wherein TDL enables a one to one mapping from at least one of properties, methods and events of the type system to an XML schema for describing the at least one of properties, methods and events and vice versa.

36. A computing device according to claim 32, wherein TDL supports inheritance of

programming constructs.

37. A computing device according to claim 30, wherein the XML-based IDL is the wire
format of message communications relating to the service between devices of the computing
5 system.

38. A computing device according to claim 37, wherein the XML-based IDL enables a one to
one mapping from the wire format to the message communications and vice versa.

39. A computing device according to claim 31, wherein TDL enables the transfer of a service
reference across an application boundary.

40. A computing device according to claim 30, wherein the computing system is a peer to
peer distributed computing environment.

41. A computing device according to claim 30, wherein the mapping mechanism for the
XML-based IDL is extendable to map additional constructs of a richer type system to an XML
schema and vice versa.